A N AUTOMATEI) MAPPING PROCESSOR USING C-BANI) INTERFER OMETRIC SAR I)ATA

E. Rodriguez, T.R. Michel, J.M. Martin, B. Houshmand
Jet Propulsion Laboratory
California Institute of Technology

Abstract: We. present the description of a processor which has been implemented to generate map products starting from C-band interferometric data. The first stage oft he processor consists of the conventional interferometric SAR processing producing a Digital 1 ilevation Model (DEMs) and a SAR bright ness image in sensor coordinates. In the second stage of processing, a land use classification map is obtained by using the DEM, brightness, and interferometric correlation layers. Auxi liary layers which include a drainage layer, a height gradient layer, a height error layer, an cst imated penet rat ion layer, and a shaded relief layer are also computed, 1 n the final step, a] 1 UTM co] locat ed la yers are combi ned in a GIS syst em which a] lows for both hard copy map products and for digital appl i cat ions.

1. I NTRODUCTION

Conventional generation of map products using stereo imaging techniques requires human i ntervent ion, making the elaborat ion of wide maps coveri ng very ext ensive areas a daunt ing proposit ion. Recent] y, Interferometric SAR (IFSAR) techniques have shown the capabilit y of producing digital elevat ion models (DEM) in near real time, and without operator intervention. Given the promise shown by these techniques, it is a natural extension to try to implement a processing system capable of generat ing digital map products autot nat icall y in near real time. The purpose of this paper is to report on such a prototype system which has been implemented at JPL for the. processing: of AIRSAR TOPSAR C-band i nterferometric data.

N7R.

The gods in developing the automated mappi ng station described here were the fol lowing:

1) Automated product generat ion without operator intervention or editing.

2) Near real time processing capabilities.

3) 1 and classification from interferometric data alone.

4) 1)igital products in a GIS usable format.

5) Abi I it y to prod uce hard copy output of standard map products.

The following section describes the Jf'1. AUTOMAP processor which has been i mplemented to achieve these goals.

2. PROCESSOR IMPI ,1WIILN'1'A'1'1ON

The AUTOMAP processon akcs raw interferometric data (e.g., the data generated by t he JPL TOPSAR radar), and produces two types of products:]) Geolocated topographic maps; and, 2) A nmlti-layer GIS database which can be examined on a work station. The AUTOMAP processor can be divided functionally into four (see Figure 1):

- The 1 I 'PROC Processor: Converts raw interferometric data into heights, radar brightness, and correlation files.

- GIS Data Layer Generation: Takes the outputs from 1 t 'PROC and computes the layers which will be incorporated into the GIS database,

- Geolocated GIS Quad: Given the desired map location and size, creates a geolocated database **incorporating** the **desired** data layers.

- Map 1 'roduct Generation: Maims map products suitable for printing,

The IFPROC processor has been documented elsewhere, and this manual will restrict itself to a brief overview of how it is integrated to the rest of the AUTOMAP processor.

The IFPROC module of the processor **always** outputs four data files which correspond to the height, brightness, correlation, and leader-data. Since user requirements for the output map products are variable, the rest of th AUTOM AP processor Idesigned for flexibility in both the output products and **in t** he content**s** of the GIS database. **The** user **can** specify 1 he desired products by using a menu, which can be filled interactively or preset prior to run time.

The processor is currently implemented in a l 6-processor Silicon Graphics Power Challenge computer and a DEC Alpha work station which is used as a front-end for displaying the GIS data and printing output products. The computer **intensive code is** run in **the** Silicon Graphics computer and is written in C++, C, and FORTRAN. The commercially available package PCI is used to geolocate the data layers, do some processing steps, and display the data. PCI processing steps are run automatically using the PCI "easi" script language.

Terrain Classification

The terrain cl ossification module takes in the height, brightness, and correlation data produced **by** 1 FPROC, computes a series of data feature **layers,** and performs **terrain classification** using **a mixture of** Bayes **classification** and knowledge based **algorithms,** A flow diagram is presented in Figure 2.

The following is a brief description of each oft he programs in the terrain **classification** module:

- featureVector: this program takes the three files generated by 1 i 'PROC anti generates a set of user defined features to be used by the Bayes classifier. The features currently implemented arc:
    - The radar brightncss corrected for incidence angle, ante.nna pattern, and range effects.
    - The measured correlation between the two 1 FSAR channels.
    - The calculated ``penetration" (correlat**ion** corrected for thermal noise and tilt effects). To obtain this feature, define the volumetric decorrelation, γ_Z, as

$$\gamma_Z = \frac{\gamma}{\gamma_G \gamma_N}$$

where γ is the measured correlation; γ_N is the decorrelation due **to** noi**sc,** defined **as**

$$\gamma_N = \frac{1}{1+ SNR^{-1}}$$

where SNR is the signal-to-noise ratio; and, γ_G is the decorrelation due to the size of the resolution cel1 and the local surface slope. The "penetration", σ_Z, which is a measure of the standard deviation of scatterers in the vertical direction, is defined as

$$\sigma_Z = \frac{2}{\kappa_Z}\sqrt{1 - \gamma_Z}$$

•The height rms inside a window whose size is determined by the user.
•The slope rms inside a window whose. size is determined by the user.
•The brightness rms inside a window whose size is determined by the user.
•1 'hc rms of the detrended Slope.
•The rms of the detrended height.
in addition, **featureVector** calculates the surface slopes in the north and east directions.

•**bayesClass:** This module implements a standard Bayes classifier assuming Gaussian probability densit y functions and equal *a priori* probabilities. The inputs consist of the features calculated by **featureVector** and a file which contains the training information required by t hc Bayes Classifier. '1 'hc philosoph y bch ind this approach has been to decompose the true feature pdfs using Gaussian basis functions, and each basis function is treal ed as a separate class in the cl assificat ion. Forth is version of the processor, a tot al of four land usc classes have been chosen: water, trees, urban, non-treed/n on-urban terrain. '1'0 go from the many (typical] y on the order of 20) classes used in t hc Bayes classification to these four classes, a projection method is used, as described below. 1'rai ning was accompli shed by using silts for which ground truth had been manuall y collected. 1 'hc training set spanned a variet y of Site.s including areas around t hc San Francisco Bay area, Crater Lake, Washington, and the].os Angeles Count y area.

•**collapseCla ss:** As discussed above, the number of classes generated by **bayesClass** is greater than the four classes ultimately used in the terrain classification layer. **collapseClass uses a** simple mapping to collapse the classification layer into four classes.

•**clean Class:**This program uses contextual information to clean up the water layer in the classificat ion. The information used i ncludes surface slope and area covered by connected Component s.

•**sieve:** In genera], the classification will have a greater resolution than is desired in the output map product. The **sieve** module decomposes the classification map into connected components and iterat ively merges smal ler ones into neighboring larger ones, unt i I every component in the classificat ion map has a minimum area."1 'his procedure avoids the blockiness associated with simple averaging algorithms anti preserves edges.

Feature Calculation

1 'hc AUTOM Al' processor current 1 y i mplements only two automat ical l y derived feat ure layers: t hc mountain/peak and drainage layers.

•**Moun tain/Peak Layer:** This layer is generated by two programs, **mountain** and **peak**. **mountain** computes a mountain mask based on slope and covered area criteria. This mask, together with the elevation, is used by peak to find all the major mountain peaks contained inside the mountain mask

- **Drainage Layer:** Duc to its large memory rcquircmcnts, the drainage layer is calculated at the. map quad level, rather than for the strip map dat a. The drai nage net work is calculated by a module of the TOPAZ topography analysis package, version 1.1 0, produced by Martz and Garbrecht for the US Department of Agriculture. The complete TOPAZ package can measure topographic properties, define surface drainage, delimit watershed boundaries, quantify the drainage network, and paramet rize subcat chments and overland flow paths, but wc use onl y a small subset of its capabilities.

1 'o obtain a drainage network, t hc DEDNM module first preprocesses the DEM to remove localized depressions by filling, them in up to the lowest outflow point, and lowering the 1)EM levels of local ized i mpoundments (dams) across drainage paths to allow the flow to continue. 1 'lat regions either in the DEM or produced by depression fill-in are the.n modified by adding small height variations consistent with the surrounding terrain and just large enough to define a flow path across the regions. The D8 (deterministic eight-neighbor) method is used to calculate the steepest descent direction for each pixel. '1'hen, st arti ng at each 1>1 EM ccl 1 in turn, the flow vectors arc fo] lowed from ccl 1 to cel l until the edge of t hc DEM is encountered, and t hc upstream area value for each ccl 1 along the path is i ncremented. After all 1)1 EM cells have been used as starling points, the upstream area value for each cell is the number of DEM cells that di ain into that ccl]. Finally, to define the drainage network, onl y cells that have an upstream area greater than a threshold val uc (t hc Critical Source Area, CSA) are marked, and paths shorter than a minimum length (Minimum Source Channel Length, MSCL) are pruned. The CSA and MSCI, can be spat ial ly varying to account for varying hydrologic cent rols, such as soi l t ypc or vegetation, but wc use only a single value for these quantities for the drainage network calculation.

'1 'hc 1)1 EDNM module has been slight l y modified fort hc AUTOM A}' pm-pose. The original program did not have an exit path after the drainage network was calculated, so one was added as an option. The DEM input routine was also modified to read a binary DEM file, and to accomodate data values less than 1 m. Finally, the output is reported as a vect or layer, rather than a byte image.

Auxiliary Layers

In addition to the previous layers, AUTOMAP computes a series Of useful auxiliary layers which may bc incorporated in the GIS data bas and output map products:

- **1 leight Error Layer:** A layer containing the random DTE height error, as derived from t hc measured correlat ion fund ion, is also calculated. The height error is calculated by first calculating the estimated phase mist, σ_Φ, using the Cramer-Rao bound

$$\sigma_\Phi = \frac{1}{\sqrt{2N}} \frac{\sqrt{1 - \gamma^2}}{\gamma}$$

where N is t hc number of looks. The height error is then estimated usi ng the formul a

$$\sigma_h = \frac{\lambda \mathbf{r} \left(\sin \theta + \cos \theta \tan \tau\right)}{2\pi B} \sigma_\Phi$$

where B is the component of the baseline perpendicular to the look direction, θ is the local incidence angle, r is the range to the scatterer location, τ is the surface slope in the azimuth direction, and λ is the wavelength.

•**Shaded Relief:** A shaded relief image is calculated from the slope layers, The brightness value is set by forming the dot product, $\hat{\mathbf{n}}\cdot\hat{\mathbf{r}}$, where $\hat{\mathbf{n}}$ is the local surface normal, and $\mathbf{r}$ is the unit vector in the look direction.

•**Brightness Modulated Classification RGB Layers:** Red, green, and blue image planes are generated using the brightness to set the image value and the classification to set the image hue.

• **Sigma0 Layer:** The radiometrically corrected brightness feature is used as an auxiliary layer.

• **Penetration Layer:** The penetration feature is used as an auxiliary layer.

Geolocation and GIS Database Generation

The geolocation is performed using nearest neighbor interpolation going from sensor coordinates to UTM. The GIS database generation is performed by using PCI, a commercially available GIS program. Given the user selected map size and location and a set of options, this module will resample the strip map layers generated in the previous module to a UTM map using the WGS 84 ellipsoid. It will, in addition, calculate a contour level vector layer. The output GIS data base will be stored in PCI format.

Map Product Generation

The map product generation module is performed by using PCI hardcopy map making utilities to generate the map data product selected by the user. The output is in a standard graphic format which can be output by a color printer or viewed electronically.

3. SAMPLE RESULTS

Figure 3 presents a sample output Of the AUTOMAP processor for data over the San Francisco Area. The training set for this classification was obtained by selecting small representative samples for each of the four classes from the data itself. in the image, Golden Gate Park and the Presidio region, which are known to be treed, are clearly recognizable. The urban areas agree with with USGS 7.5' quads. The Bay and Pacific ocean are also clearly identified. A detailed comparison against ground truth confirms many Of the detailed features observed in the classification map, although there are areas, such as in the hills above Golden Gate Park where the extent of tree coverage is over-estimated. The contour lines shown in the map are obtained from the TOPSAR IFSAR derive.d DEM.

4. CONCLUSIONS

WE have presented a description of the JPL AUTOMAP processor for generation of map products and auxiliary layers using C-band IFSAR data alone. Preliminary results indicate that the techniques presented here represent an automated alternative to traditional stereo photography for generating finished map products. Furthermore, the current processing speed of the AUTOMAP processor is 3 squared kilometers/per minute using a 16 processor Silicon Graphics Power Challenge. This efficiency is much greater than that which can be achieved using conventional optical techniques.

S. ACKNOWLEDGEMENTS

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under contract with the National Aeronautics and Space Administration.

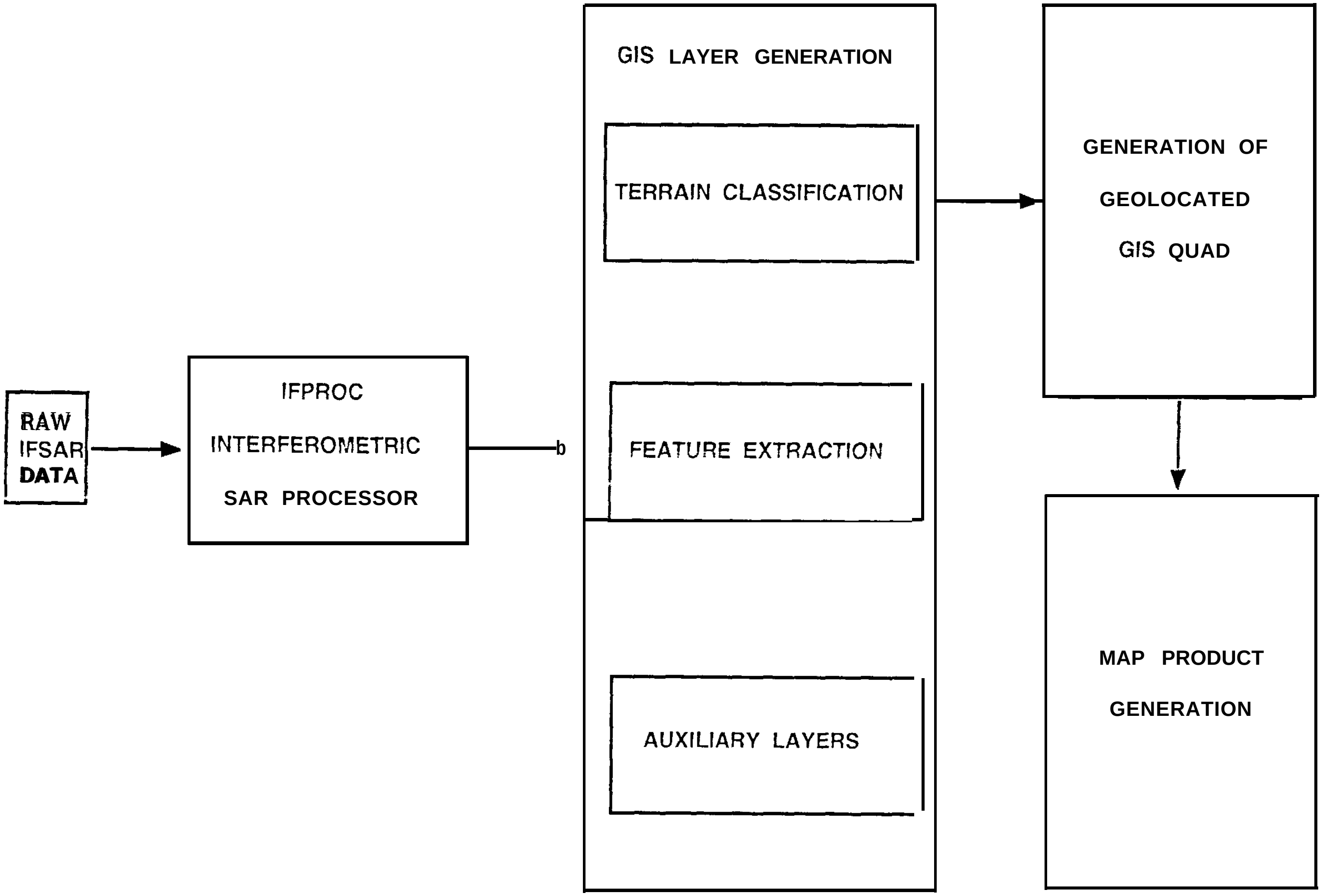

Figure 1: Flow diagram for the JPL AUTOMAP Processor

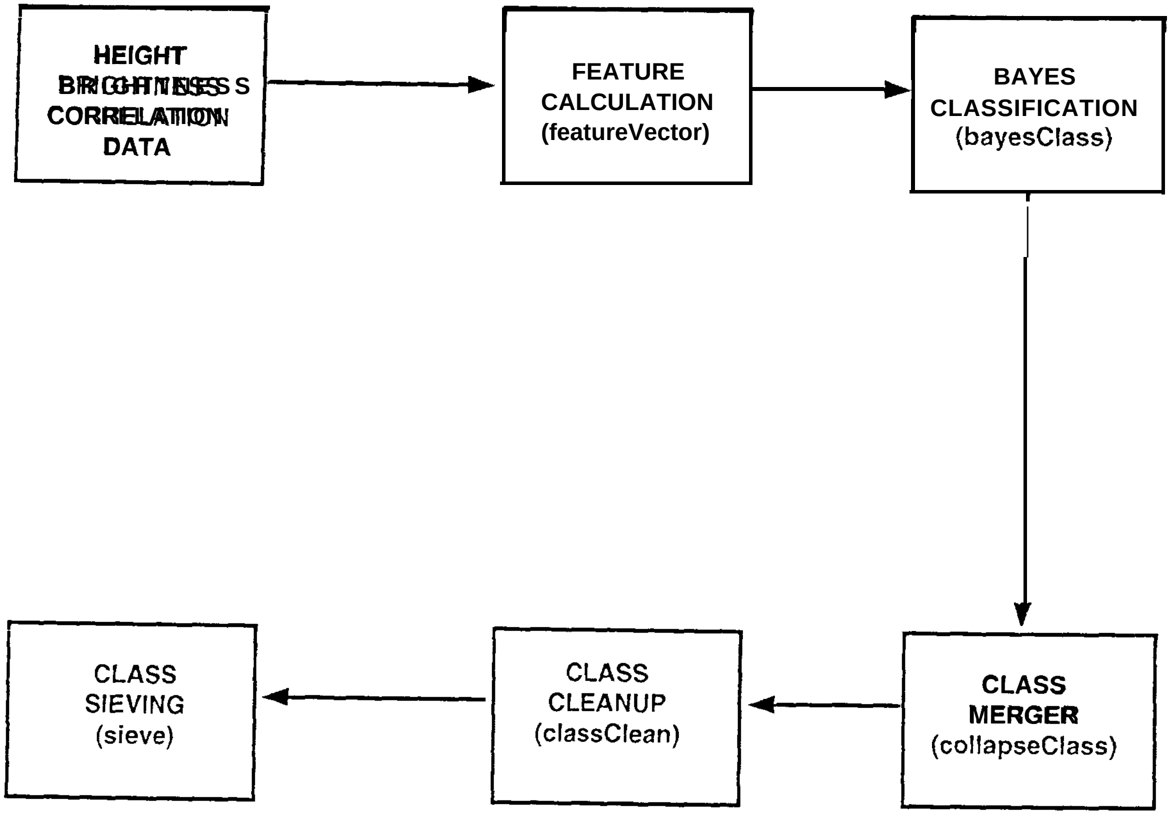

Figure 2: Flow diagram for the terrain classification module of the JPL AUTOMAP Procesor.

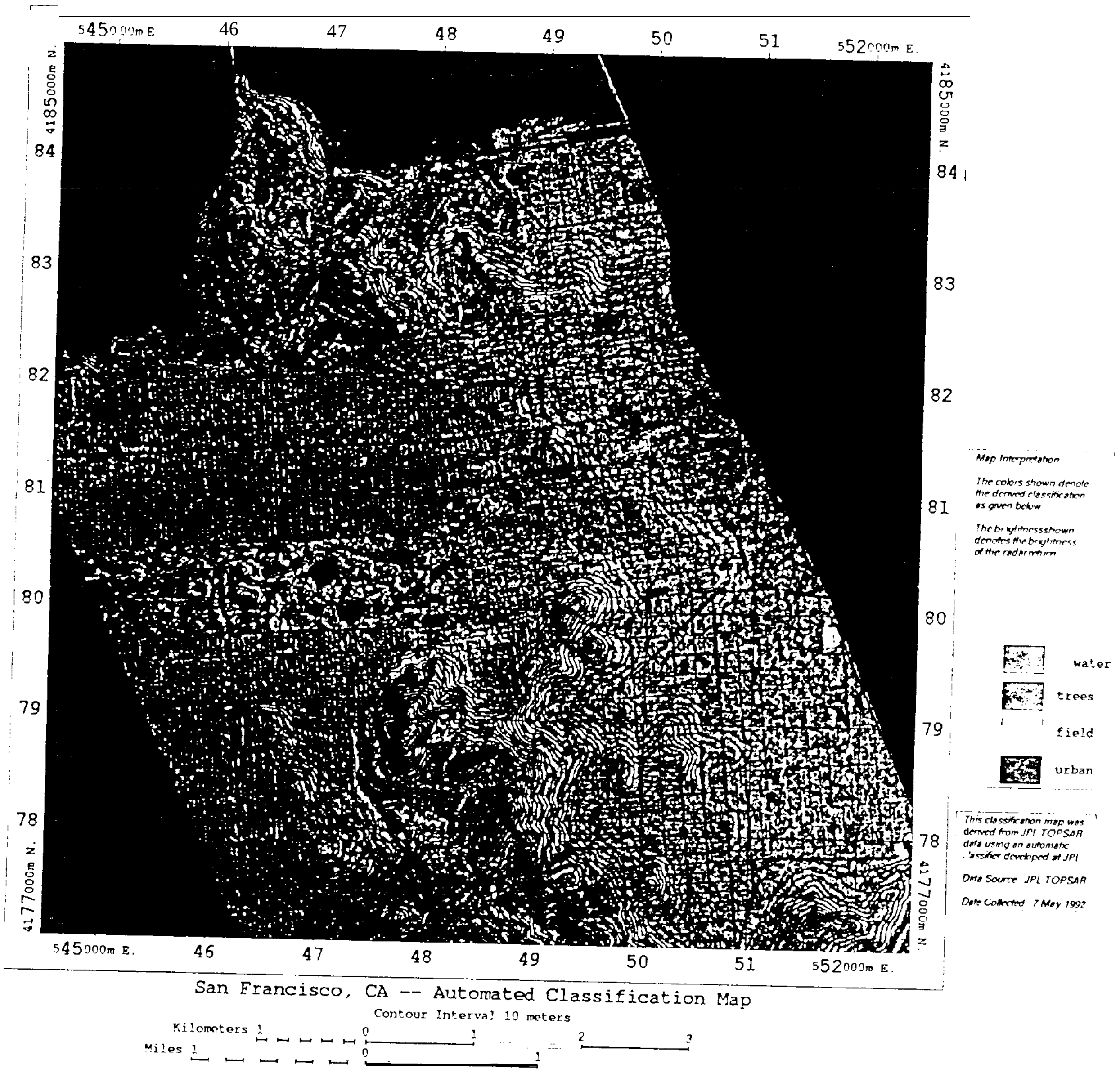
545000m E.
46
47
48
49
50
51
552000m E.
4185000m N.
84
83
82
81
80
79
78
4177000m N.
Map Interpretation
The colors shown denote the derived classification as given below
The brightness shown denotes the brightness of the radar return
water
trees
field
urban
This classification map was derived from JPL TOPSAR data using an automatic classifier developed at JPL
Data Source JPL TOPSAR
Date Collected 7 May 1992
San Francisco, CA -- Automated Classification Map
Contour Interval 10 meters
Kilometers 1
0
1
2
3
Miles 1
0
1

Figure 3: Sample product of the JPL automap processor showing IFSAR generated height contour levels, terrain classification (image hue), and radar brightness (image value) simultaneously.